CLAIM AMENDMENTS

Claim Amendment Summary

Claims pending

- At time of the Action: Claims 1-50.
- After this Response: Claims 6-18 and 30-42.

Canceled or Withdrawn claims: 1-5, 19-29, and 43-50.

Amended claims: 6-8, 12-14, 30-32 and 36.

New claims: none.

Claims:

Claims 1-5 are **CANCELED.**

6. **(PRESENTLY AMENDED)** ~~The apparatus in claim 5 wherein:~~ Apparatus for forming an identifier for an input object and for securely marking the input object with the identifier so as to yield a marked object, the apparatus comprising:

a processor; and

a memory having computer executable instructions stored therein;

the processor being configured to perform acts, in response to the stored executable instructions, comprising:

- generating a flow representation for the input object, the representation having a plurality of nodes, said nodes representing pre-defined first

421 West Riverside, Suite 500 Spokane, WA 99201 P: 509.324-9256 F: 509.323-8979 www.leehayes.com
lee&hayes

operations performed by the input object, and connections among the nodes signifying associated flow among the pre-defined first operations performed by the input object;

- randomly selecting first and second nodes from the plurality of nodes in the representation so as to form a pre-defined number of nodal pairs, each of said pairs having one of the first nodes and a corresponding one of the second nodes;
- for each of the nodal pairs, forming an executable procedure associated with each of the nodal pairs by establishing flow between the first and second nodes in said each nodal pair and inserting, in the flow so established, a selected one of a plurality of different pre-defined second operations so as to collectively define the marked object, whereby the marked object implements the pre-defined first operations and a plurality of selected ones of the pre-defined second operations, each of which has been randomly spliced into flow of the input object, wherein the identifier collectively comprises all ones which differ within the plurality of pre-defined second operations, and execution flow associated therewith and involving the nodal pairs;
- inserting a pre-defined number of separate links and designations for selected ones of the procedures into the flow representation so as to yield a combined flow representation; and
- converting, in response to said input executable code and executable code for said selected ones of the procedures, said combined flow representation into output executable code, said output executable code being the marked code;

421 West Riverside, Suite 500
Spokane, WA 99201
P: 509.324-9256
F: 509.323-8979
www.leehayes.com
lee&hayes

wherein:

- the input object comprises a software object, which comprises input executable code, at least one instruction in the input executable code being associated with a corresponding one of the pre-defined first operations, and executable code for a corresponding executable procedure being associated with each selected one of the pre-defined second operations;
- the input executable code comprises first and second portions thereof and the flow representation comprises first and second separate flow representations for the first and second portions of the input executable code, respectively;
- the first portion of the input executable code comprises at least a pre-defined portion of a non-marked application program; and
- the second portion of the input executable code comprises a remaining portion of the non-marked application program or pre-defined executable security code.

7. **(PRESENTLY AMENDED)** The apparatus in claim 6 wherein the processor, in response to the stored instructions, inserts executable code for ~~the~~ a selected one procedure in noncontiguous locations in the input executable code.

8. **(PRESENTLY AMENDED)** The apparatus in claim 6 wherein the processor, in response to the stored instructions, selects ~~the~~ a procedure from a pre-defined library of stored routines, wherein said procedure is one of the stored routines.

421 West Riverside, Suite 500
Spokane, WA 99201
P: 509.324-9256
F: 509.323-8979
www.leehayes.com
lee&hayes

9. **(ORIGINAL)** The apparatus in claim 8 wherein each of the inserted procedures implements, when executed, a pre-defined function such that if any of said inserted procedures is removed from the marked code, the marked code, when subsequently executed, will terminate its execution.

10. **(ORIGINAL)** The apparatus in claim 8 wherein at least one of the inserted procedures implements, when executed, a pre-defined function which is independent of functionality provided by the non-marked application program.

11. **(ORIGINAL)** The apparatus in claim 6 wherein the security code provides functionality independent of any functionality provided by the application program.

12. **(PRESENTLY AMENDED)** The apparatus in claim 6 wherein the processor, in response to the stored instructions:

(a) generates first and second separate flow representations for the first and second portions of the input executable code;

(b) partitions each of the first and second flow representations into k clusters each so as to yield first and second cluster flow representations, respectively (where k is a pre-defined integer);

(c) randomly selects the first and second nodes in the first and second cluster flow representations, respectively, so as to form a corresponding one of the nodal pairs;

421 West Riverside, Suite 500
Spokane, WA 99201
P: 509.324-9256
F: 509.323-8979
www.leehayes.com
lee&hayes

(d) inserts a designation for ~~the~~ a selected executable procedure at a first node in the nodal pair; and

(e) repeats operations (c) and (d) a pre-defined number of times so as to insert a pre-defined number of separate procedures into the first and second flow representations so as to yield the combined flow representation.

13. (PRESENTLY AMENDED) The apparatus in claim 12 wherein the processor, in response to the stored instructions, inserts executable code for ~~the~~ a selected one procedure in noncontiguous locations in the input executable code.

14. (PRESENTLY AMENDED) The apparatus in claim 12 wherein the processor, in response to the stored instructions, selects ~~the~~ a one procedure from a pre-defined library of stored routines, wherein said one procedure is one of the stored routines.

15. (ORIGINAL) The apparatus in claim 14 wherein each of the inserted procedures implements, when executed, a pre-defined function such that if any of said inserted procedures is removed from the marked code, the marked code, when subsequently executed, will terminate its execution.

16. (ORIGINAL) The apparatus in claim 14 wherein at least one of the inserted procedures implements, when executed, a pre-defined function which is independent of functionality provided by the non-marked application program.

421 West Riverside, Suite 500
Spokane, WA 99201
P: 509.324-9256
F: 509.323-8979
www.leehayes.com
lee@hayes

17. (ORIGINAL) The apparatus in claim 12 wherein the processor, in response to the stored instructions:

randomly selects a node, U, in the first cluster flow representation;

randomly selects, with probability 1 - λ (where λ is a pre-defined value with $0 \leq \lambda \leq 1$), a node Y in the second cluster flow representation;

randomly selects, with probability λ, a node Z, other than U, in the first cluster flow representation; and

provides designations of nodes Y and Z as the nodes forming the nodal pair.

18. (ORIGINAL) The apparatus in claim 12 wherein the processor, in response to the stored instructions, randomly selects the first and second nodes from different clusters solely within the first cluster flow representation or from different clusters solely within the second cluster flow representation.

Claims 19-29 are CANCELED.

30. (PRESENTLY AMENDED) ~~The method in claim 29 wherein:~~ For use with a computer system having a processor and a memory, the memory having computer executable instructions stored therein, a method for forming an identifier for input executable code and for securely marking the input executable code with the identifier so as to yield marked code, the method comprising the steps of:

generating a flow representation for the input object, the representation having a plurality of nodes, said nodes representing pre-defined first operations performed by the input object, and connections among the nodes signifying

421 West Riverside, Suite 500
Spokane, WA 99201
P: 509.324-9256
F: 509.323-8979
www.leehayes.com
lee&hayes

associated flow among the pre-defined first operations performed by the input object;

randomly selecting first and second nodes from the plurality of nodes in the representation so as to form a pre-defined number of nodal pairs, each of said pairs having one of the first nodes and a corresponding one of the second nodes;

for each of the nodal pairs, forming an executable procedure associated with each of the nodal pairs by establishing flow between the first and second nodes in said each nodal pair and inserting, in the flow so established, a selected one of a plurality of different pre-defined second operations so as to collectively define the marked object, whereby the marked object implements the pre-defined first operations and a plurality of selected ones of the pre-defined second operations, each of which has been randomly spliced into flow of the input object, wherein the identifier collectively comprises all ones which differ within the plurality of pre-defined second operations, and execution flow associated therewith and involving the nodal pairs;

inserting a pre-defined number of separate links and designations for selected ones of the procedures into the flow representation so as to yield a combined flow representation; and

converting, in response to said input executable code and executable code for said selected ones of the procedures, said combined flow representation into output executable code, said output executable code being the marked code;

wherein:

- the input object comprises a software object, which comprises input executable code, at least one instruction in the input executable code being associated with a corresponding one of the pre-defined first

421 West Riverside, Suite 500
Spokane, WA 99201
P: 509.324-9256
F: 509.323-8979
www.leehayes.com
lee@hayes

operations, and executable code for a corresponding executable procedure being associated with each selected one of the pre-defined second operations;

- the input executable code comprises first and second portions thereof and the flow representation comprises first and second separate flow representations for the first and second portions of the input executable code, respectively;
- the first portion of the input executable code comprises at least a pre-defined portion of a non-marked application program; and
- the second portion of the input executable code comprises a remaining portion of the non-marked application program or pre-defined executable security code.

31. **(PRESENTLY AMENDED)** The method in claim 30 further comprising the step of selecting ~~the~~ a procedure from a pre-defined library of stored routines, wherein said procedure is one of the stored routines.

32. **(PRESENTLY AMENDED)** The method in claim 31 further comprising the step of inserting executable code for ~~the~~ a selected one procedure in noncontiguous locations in the input executable code.

33. **(ORIGINAL)** The method in claim 31 wherein each of the inserted procedures implements, when executed, a pre-defined function such that if any of said inserted procedures is removed from the marked code, the marked code, when subsequently executed, will terminate its execution.

421 West Riverside, Suite 500
Spokane, WA 99201
P: 509.324-9256
F: 509.323-8979
www.leehayes.com
lee@hayes

34. **(ORIGINAL)** The method in claim 31 wherein at least one of the inserted procedures implements, when executed, a pre-defined function which is independent of functionality provided by the non-marked application program.

35. **(ORIGINAL)** The method in claim 30 wherein the security code provides functionality independent of any functionality provided by the application program.

36. **(PRESENTLY AMENDED)** The method in claim 30 further comprising the steps of:

(a) generating first and second separate flow representations for the first and second portions of the input executable code;

(b) partitioning each of the first and second flow representations into k clusters each so as to yield first and second cluster flow representations, respectively (where k is a pre-defined integer);

(c) randomly selecting the first and second nodes in the first and second cluster flow representations, respectively, so as to form a corresponding one of the nodal pairs;

(d) inserting a designation for ~~the~~ a selected executable procedure at a first node in the nodal pair; and

(e) repeating operations (c) and (d) a pre-defined number of times so as to insert a pre-defined number of separate procedures into the first and second flow representations so as to yield the combined flow representation.

421 West Riverside, Suite 500
Spokane, WA 99201
P: 509.324-9256
F: 509.323-8979
www.leehayes.com
lee&hayes

37. **(ORIGINAL)** The method in claim 36 further comprising the step of inserting executabl code for the selected one procedure in noncontiguous locations in the input executable code.

38. **(ORIGINAL)** The method in claim 36 further comprising the step of selecting the procedure from a pre-defined library of stored routines, wherein said procedure is one of the stored routincs.

39. **(ORIGINAL)** The method in claim 38 wherein each of the inserted procedures implements, when executed, a pre-defined function such that if any of said inserted procedures is removed from the marked code, the marked code, when subsequently executed, will terminate its execution.

40. **(ORIGINAL)** The method in claim 38 wherein at least one of the inserted procedures implements, when executed, a pre-defined function which is independent of functionality provided by the non-marked application program.

41. **(ORIGINAL)** The method in claim 36 wherein the first and second nodes randomly selecting step comprises:

randomly selecting a node, U, in the first cluster flow representation;

randomly selecting, with probability 1 - λ (where λ is a pre-defined valuc with $0 \leq \lambda \leq 1$), a node Y in the second cluster flow representation;

randomly selecting, with probability λ, a node Z, other than U, in the first cluster flow representation; and

421 West Riverside, Suite 500
Spokane, WA 99201
P: 509.324-9256
F: 509.323-8979
www.leehayes.com
lee&hayes

providing designations of nodes Y and Z as the nodes forming the nodal pair.

42. **(ORIGINAL)** The method in claim 36 wherein the first and second nodes randomly selecting step comprises the step of randomly selecting the first and second nodes from different clusters solely within the first cluster flow representation or from different clusters solely within the second cluster flow representation.

Claims 43-50 are **CANCELED**.

421 West Riverside, Suite 500
Spokane, WA 99201
P: 509.324-9256
F: 509.323-8979
www.leehayes.com
lee&hayes